

**Listing of Claims:**

1. (Previously presented) A method comprising:

for a first logical processor, obtaining a lock on a semaphore controlling exclusive access to a resource descriptor, the resource descriptor describing a usage allocation of resources shared among a plurality of logical processors;

obtaining exclusive access for said first logical processor to said resource descriptor if said lock is obtained;

querying said resource descriptor to determine whether resources needed by said first logical processor are available;

if resources needed by said first logical processor are available, updating said resource descriptor to reserve said resources for exclusive use by said first logical processor; and

releasing said exclusive access for said first logical processor to said resource descriptor.

2. (Original) The method of claim 1, further comprising:

if said resources needed by said first logical processor are not available, releasing said exclusive access for said first logical processor to said resource descriptor.

3. (Original) The method of claim 1, further comprising, after the releasing, accessing a shared resource by said first logical processor.

4. (Original) The method of claim 1, further comprising:

after exclusive access for said first logical processor to said resource descriptor is released, obtaining exclusive access for a second logical processor to said resource descriptor;

querying said resource descriptor to determine whether resources needed by said second logical processor are available;

if resources needed by said second logical processor are available, updating said resource descriptor to reserve said resources for the exclusive use of said second logical processor; and

releasing said exclusive access for said second logical processor to said resource descriptor.

5. (Original) The method of claim 4 further comprising:

if said resources needed by said second logical processor are not available, releasing said exclusive access for said second logical processor to said resource descriptor.

6. - 9. (Canceled)

10. (Previously presented) An apparatus comprising:

a plurality of logical processors;  
a plurality of resources shared by said plurality of logical processors;  
a resource descriptor to identify a status of said shared resources; and  
a semaphore to reserve exclusive access for one of said plurality of logical processors to said resource descriptor.

11. (Previously presented) The apparatus of claim 10, further comprising logic to:

cause a first logical processor to update said semaphore to reserve exclusive access to said resource descriptor;

cause said first logical processor to update said resource descriptor to reserve exclusive use of at least a first resource of said shared resources; and

subsequently cause said first logical processor to update said semaphore to release said exclusive access.

12. (Previously presented) The apparatus of claim 11, said logic to further:

cause a second logical processor to update said semaphore to reserve exclusive access to said resource descriptor;

cause said second logical processor to update said resource descriptor to reserve exclusive use of at least a second resource of said shared resources;  
and

subsequently cause said second logical processor to update said semaphore to release said exclusive access;

wherein after reserving exclusive use of said first and second resources, respectively, said first and second logical processors concurrently use said first and second resources, respectively.

13. (Previously presented) A machine-readable medium storing instructions to perform a method comprising:

by a first logical processor,

setting a lock bit in a semaphore register to reserve exclusive access to a resource descriptor register;

generating a first bitmap identifying a first required resource;

applying said first bitmap to said resource descriptor register to reserve said first required resource;

re-setting said semaphore lock bit to release said exclusive access; and

using said first resource.

14. (Previously presented) The machine-readable medium of claim 13, said method further comprising:

by a second logical processor,

after said first logical processor has re-set said semaphore lock bit, setting said semaphore lock bit;

generating a second bitmap identifying a second required resource;

applying said second bitmap into said resource descriptor register to reserve said second required resource;

re-setting re-set said semaphore lock bit to release said exclusive access;

and

using said second resource;

wherein said first and second logical processors use said first and second resources in parallel.

15. (Previously presented) The machine-readable medium of claim 13, wherein said setting a lock bit comprises supplying an identifier of said first logical processor for writing into said semaphore register.

16. - 17. (Canceled)

18. (Previously presented) A system comprising:

- a plurality of logical processors;
- a plurality of resources to be shared by said logical processors;
- a resource descriptor to control access to said resources;
- a semaphore register to reserve exclusive access for one of said plurality of logical processors to said resource descriptor; and
- access control logic to allocate one or more of said shared resources only when granted exclusive access to said resource descriptor by said semaphore register.

19. (Previously presented) The system of claim 18, wherein said resource descriptor includes a plurality of fields each to associate a resource with a logical processor identifier.

20. (Previously presented) The system of claim 18, wherein said access control logic is to

- obtain a lock on said semaphore register to reserve exclusive access to said resource descriptor,

- determine whether a needed resource is available based on said resource descriptor,

- if so, reserve the resource, and

- release the lock on the semaphore register.

21. (Previously presented) The system of claim 20, wherein said access control logic is further to reserve one or more resources by assigning a logical processor identifier to a corresponding resource.

22. (Previously presented) The system of claim 18, further comprising unlock logic to prevent a failing logical processor from retaining a lock on the semaphore register.

23. (Previously presented) The system of claim 22, wherein the unlock logic includes causing a logical processor different from the failing logical processor to call a semaphore lock release routine and pass the routine the identifier of the failing logical processor.